[Thực hành] Ứng dụng kiểm tra email hợp lệ sử dụng Maven

**Mục tiêu**

Luyện tập sử dụng Controller trong Spring MVC.

**Điều kiện**

Biết cách sử dụng Controller trong Spring MVC.

**Mô tả**

Trong phần này, chúng ta sẽ xây dựng một ứng dụng cho phép kiểm tra tính hợp lệ của một email.

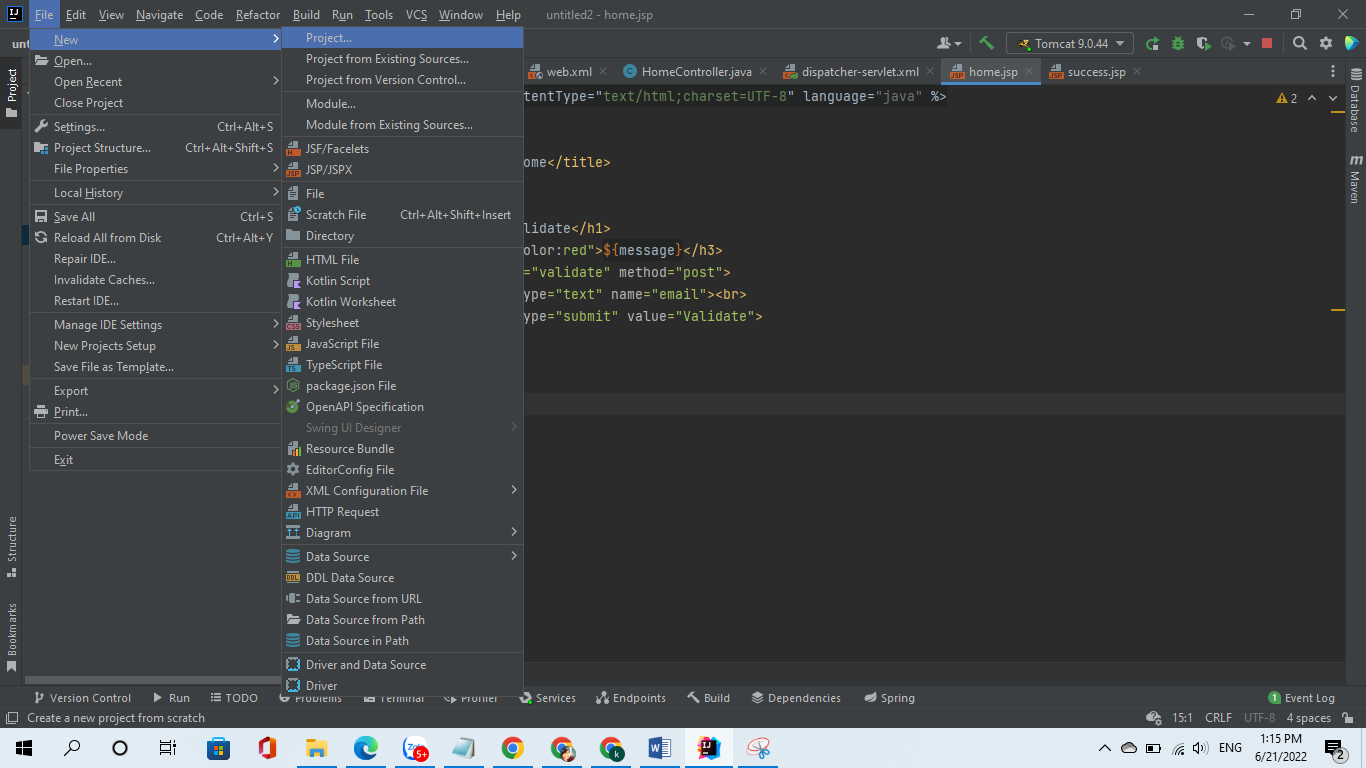
Ứng dụng cho phép người dùng nhập vào một email, nhấn nút kiểm tra và sẽ hiển thị thông báo kết quả.

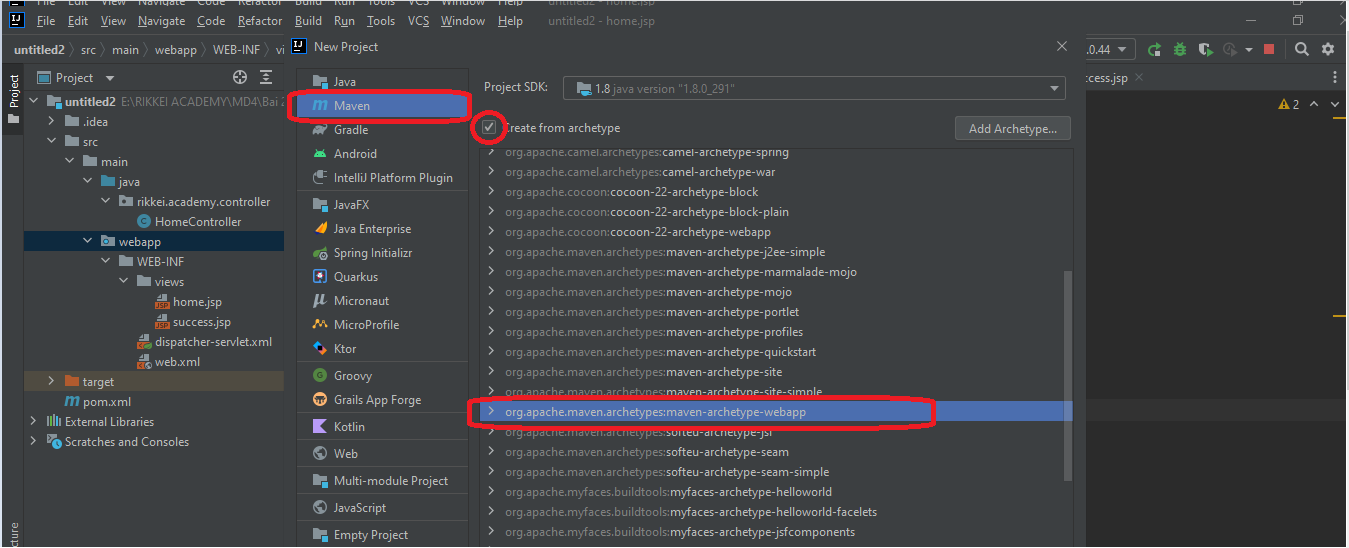
Một email hợp lệ phải đạt những yêu cầu sau:

* Bắt đầu bằng một ký tự
* Không chứa các ký tự đặc biệt
* Có một ký tự @ ngăn cách tên email và domain
* Domain phải hợp lệ

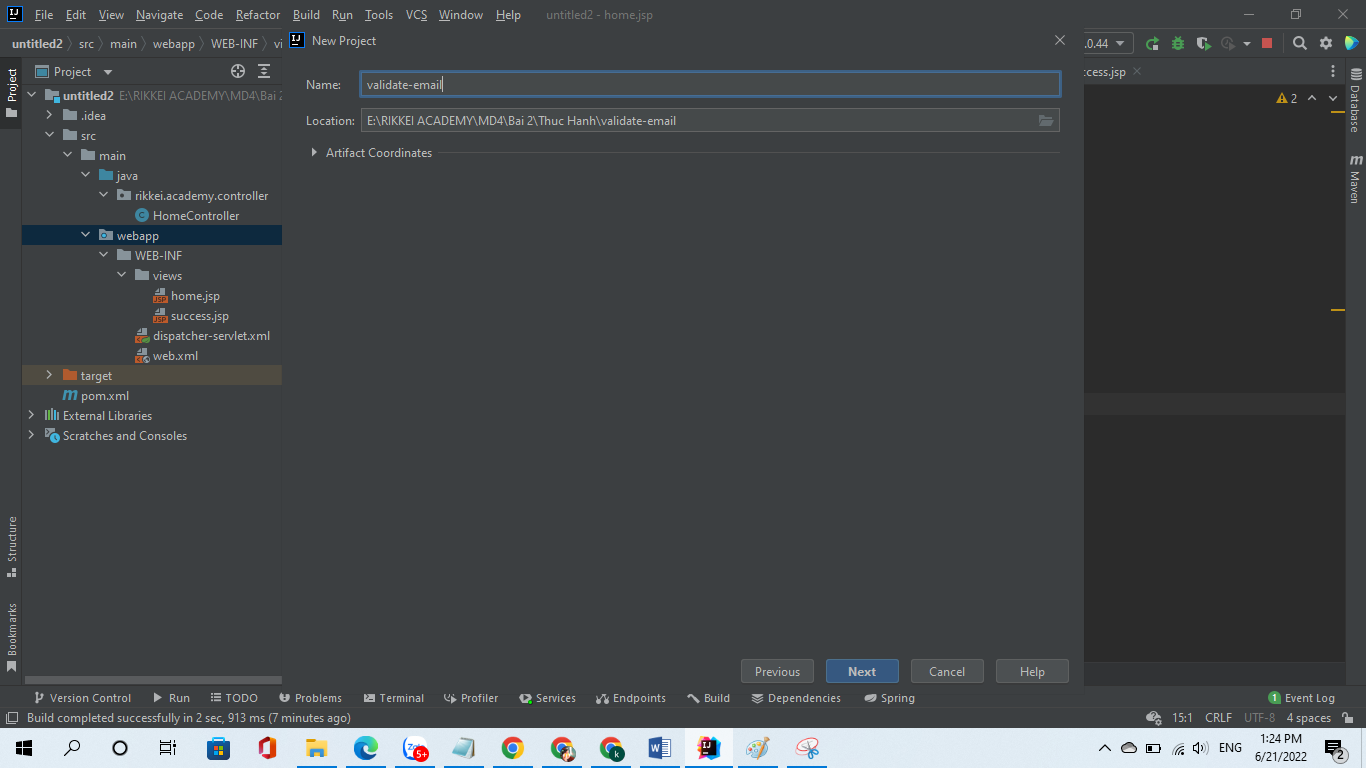
**Hướng dẫn**

* + **Tạo maven project:**



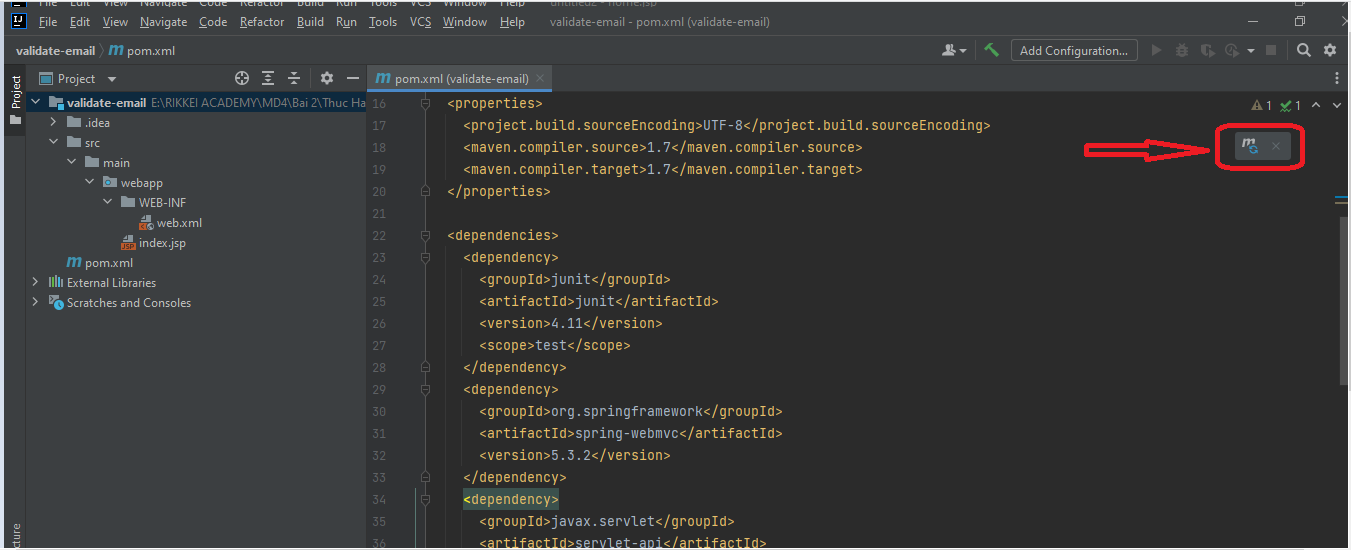


* + Đặt tên cho project:



* + Ta thêm code vào ***<dependencies></dependencies>*** trong file: **porm.xml**

<dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-webmvc</artifactId>  
 <version>5.3.2</version>  
</dependency>  
<dependency>  
 <groupId>javax.servlet</groupId>  
 <artifactId>servlet-api</artifactId>  
 <version>2.5</version>  
 <scope>provided</scope>  
</dependency>



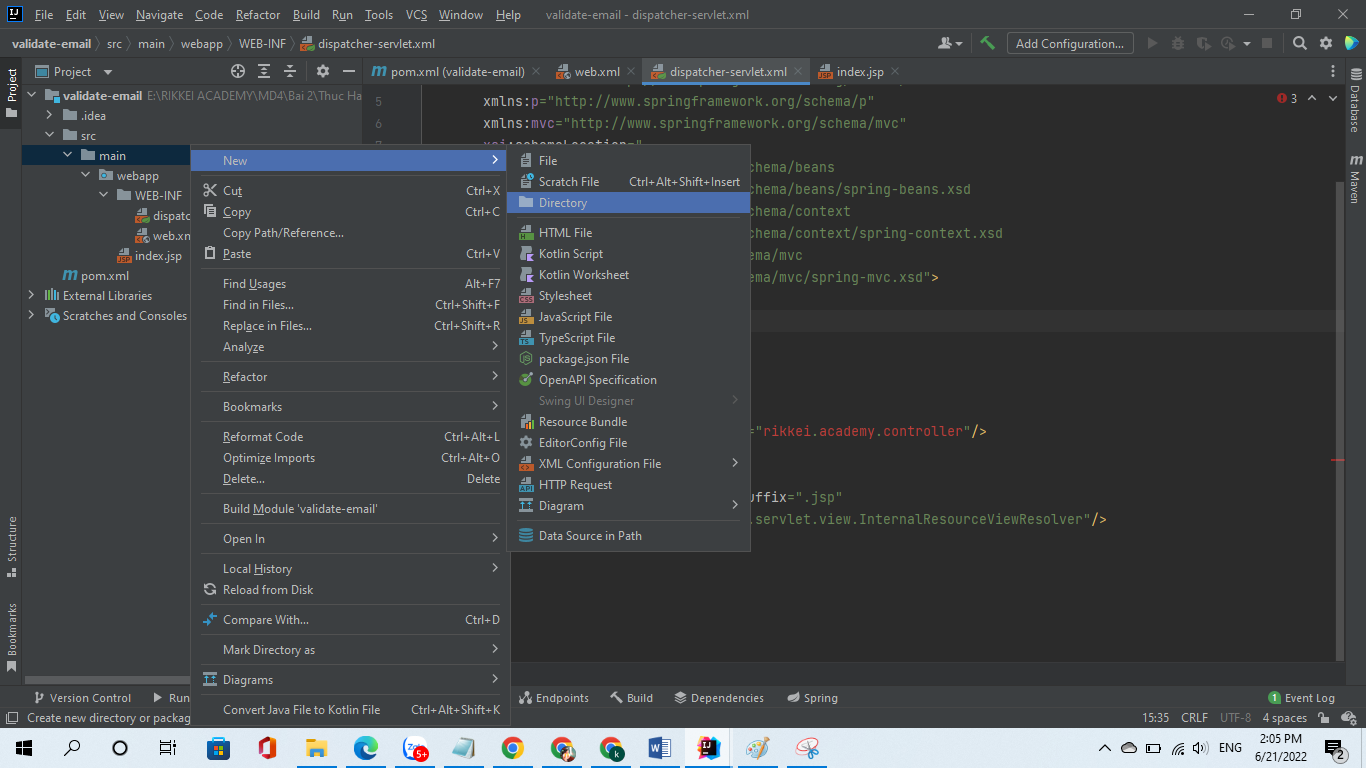
* + Nhấn vào M() => Để build thư viện
  + Sửa lại file: **web.xml**

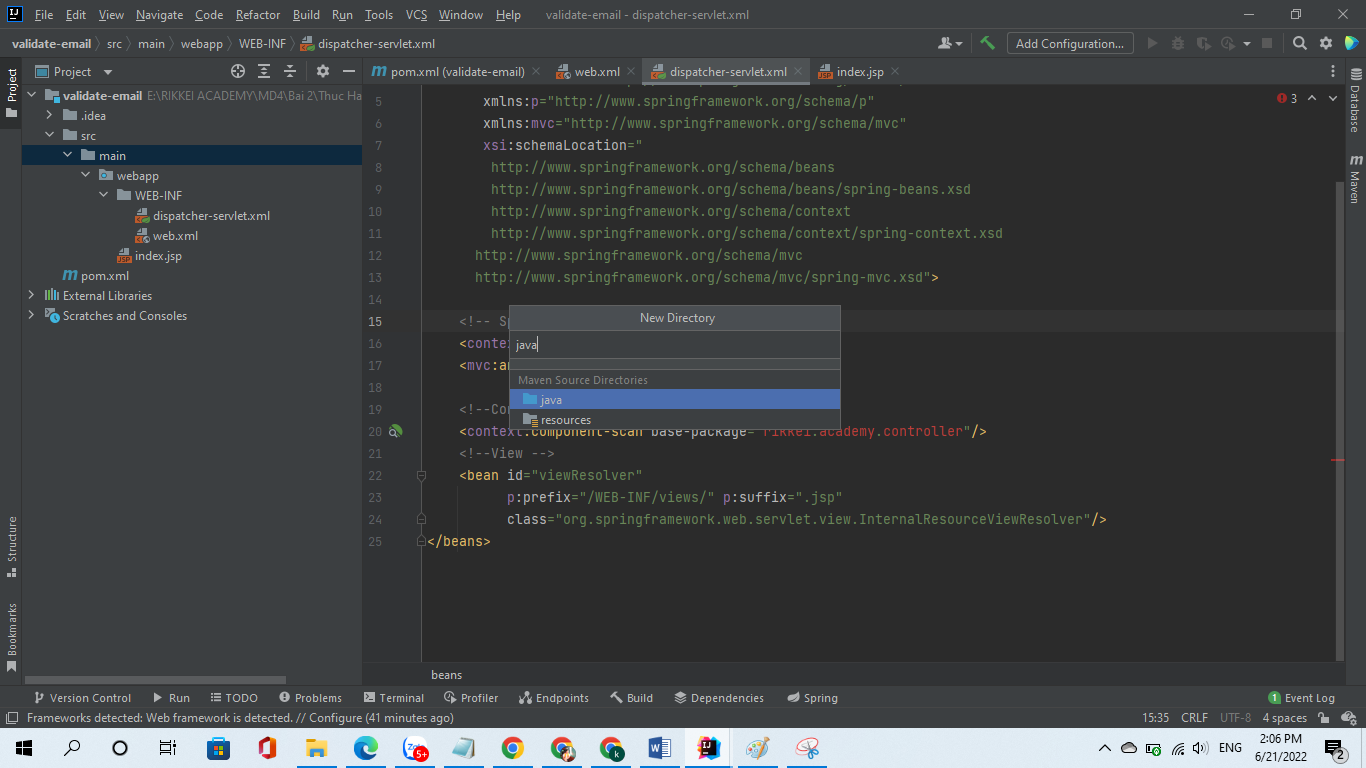
<?xml version="1.0" encoding="UTF-8"?>  
  
<web-app id="WebApp\_ID" version="2.4"  
 xmlns="http://java.sun.com/xml/ns/j2ee"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee  
 http://java.sun.com/xml/ns/j2ee/web-app\_2\_4.xsd">  
 <display-name>example</display-name>  
 <servlet>  
 <servlet-name>spring</servlet-name>  
 <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>  
 <init-param>  
 <param-name>contextConfigLocation</param-name>  
 <param-value>/WEB-INF/dispatcher-servlet.xml</param-value>  
 </init-param>  
 <load-on-startup>1</load-on-startup>  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>spring</servlet-name>  
 <url-pattern>/</url-pattern>  
 </servlet-mapping>  
</web-app>

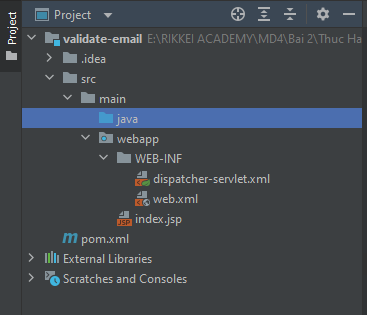
* + Tạo file: **dispatcher-servlet.xml** trong thư mục: WEB-INF : và thêm vào đoạn code config giống như các bài thực hành trước

<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:mvc="http://www.springframework.org/schema/mvc"  
 xsi:schemaLocation="  
 http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context.xsd  
 http://www.springframework.org/schema/mvc  
 http://www.springframework.org/schema/mvc/spring-mvc.xsd">  
  
 <!-- Spring MVC Annotation -->  
 <context:annotation-config />  
 <mvc:annotation-driven/>  
  
 <!--Controller -->  
 <context:component-scan base-package="rikkei.academy.controller"/>  
 <!--View -->  
 <bean id="viewResolver"  
 p:prefix="/WEB-INF/views/" p:suffix=".jsp"  
 class="org.springframework.web.servlet.view.InternalResourceViewResolver"/>  
</beans>

* + Tạo thư mục java từ main:







* + Tào package: rikkei.academy.controller từ java:
  + Tạo Class: HomeController từ package: controller
  + Thêm code vào: HomeController

package rikkei.academy.controller;  
  
import org.springframework.stereotype.Controller;  
import org.springframework.ui.ModelMap;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.PostMapping;  
import org.springframework.web.bind.annotation.RequestParam;  
  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
@Controller  
public class HomeController {  
 private static final String *EMAIL\_REGEX* = "^[A-Za-z0-9]+[A-Za-z0-9]\*@[A-Za-z0-9]+(\\.[A-Za-z0-9]+)$";  
 private static Pattern *pattern*;  
 private Matcher matcher;  
  
 public HomeController() {  
 *pattern* = Pattern.*compile*(*EMAIL\_REGEX*);  
 }  
  
 @GetMapping(value = "/")  
 public String home() {  
 return "home";  
 }  
  
 @PostMapping(value = "/validate")  
 public String user(@RequestParam("email") String email, ModelMap modelMap) {  
 boolean isValid = this.validate(email);  
 if (!isValid) {  
 modelMap.addAttribute("message", "Email is invalid");  
 return "home";  
 }  
 modelMap.addAttribute("email", email);  
 return "success";  
 }  
  
 private boolean validate(String regex) {  
 matcher = *pattern*.matcher(regex);  
 return matcher.matches();  
 }  
  
}

Trước hết chúng ta có Email Regular Expression Pattern như sau:

^[A-Za-z0-9]+[A-Za-z0-9]\*@[A-Za-z0-9]+(\\.[A-Za-z0-9]+)$;

Mô tả

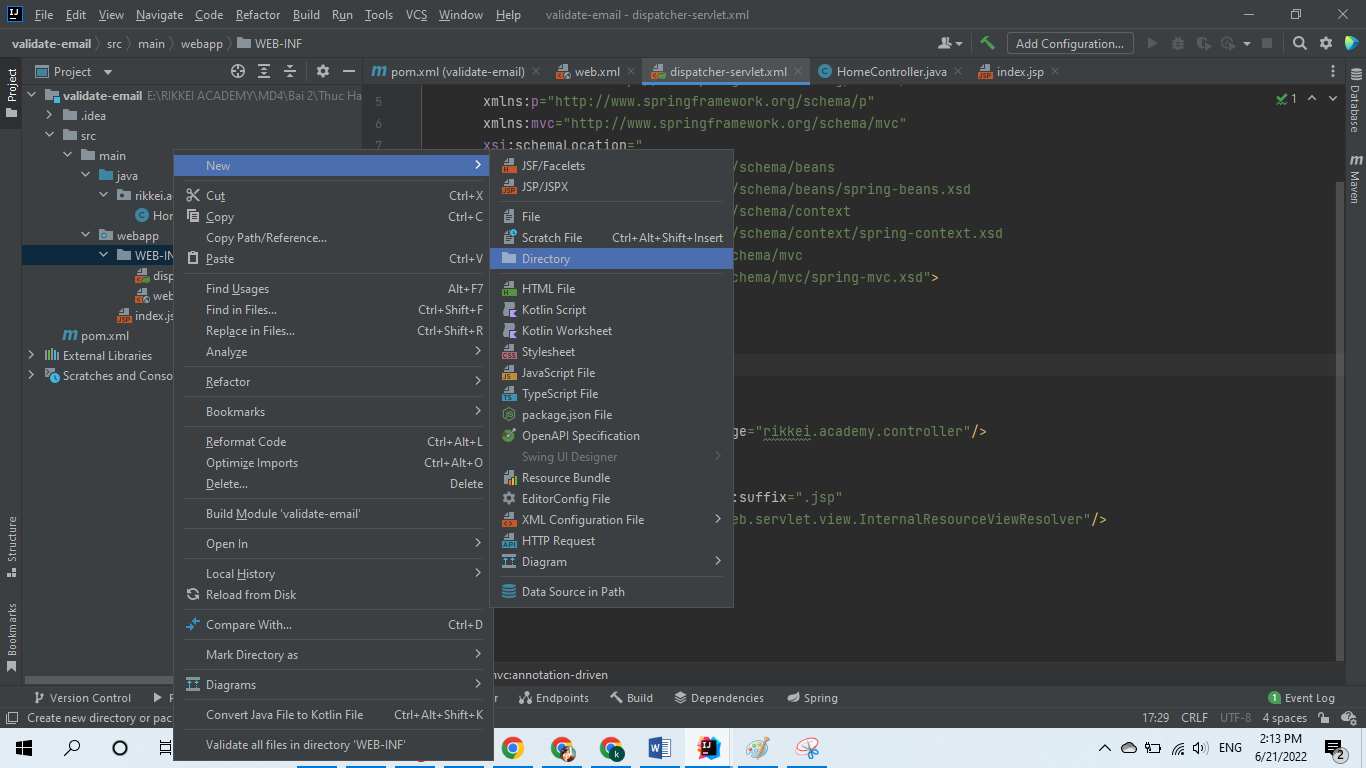
|  |  |
| --- | --- |
| Ký hiệu | Mô tả |
| ^ | Bắt đầu chuỗi String hoặc một dòng |
| [A-Za-z0-9]+ | Email address phải bắt đầu với “[A-Za-z0-9]+” và có ít nhất một ký tự |
| [A-Za-z0-9]\*@ | tùy chọn theo "[A-Za-z0-9]", và kết thúc bằng một ký hiệu "@" |
| [A-Za-z0-9] | Domain phải sau @ và ký tự phải nằm trong [A-Za-z0-9] |
| \\.[A-Za-z0-9] | Sau domain là phần mở rộng của domain sau dấu chấm, ví dụ: (.com, .net, .org) |
| $ | Kết thúc chuỗi String hoặc kết thúc dòng |

*Ở đây ta tạo một biến static String EMAIL\_REGEX chứa định dạng regular expression của email. Sau đó tạo một phương thức validate để kiểm tra email cần kiểm tra có hợp lệ với EMAIL\_REGEX đã định nghĩa hay không.*

* Ở trong file: dispatcher-servlet => Ta thấy đoạn code sau:

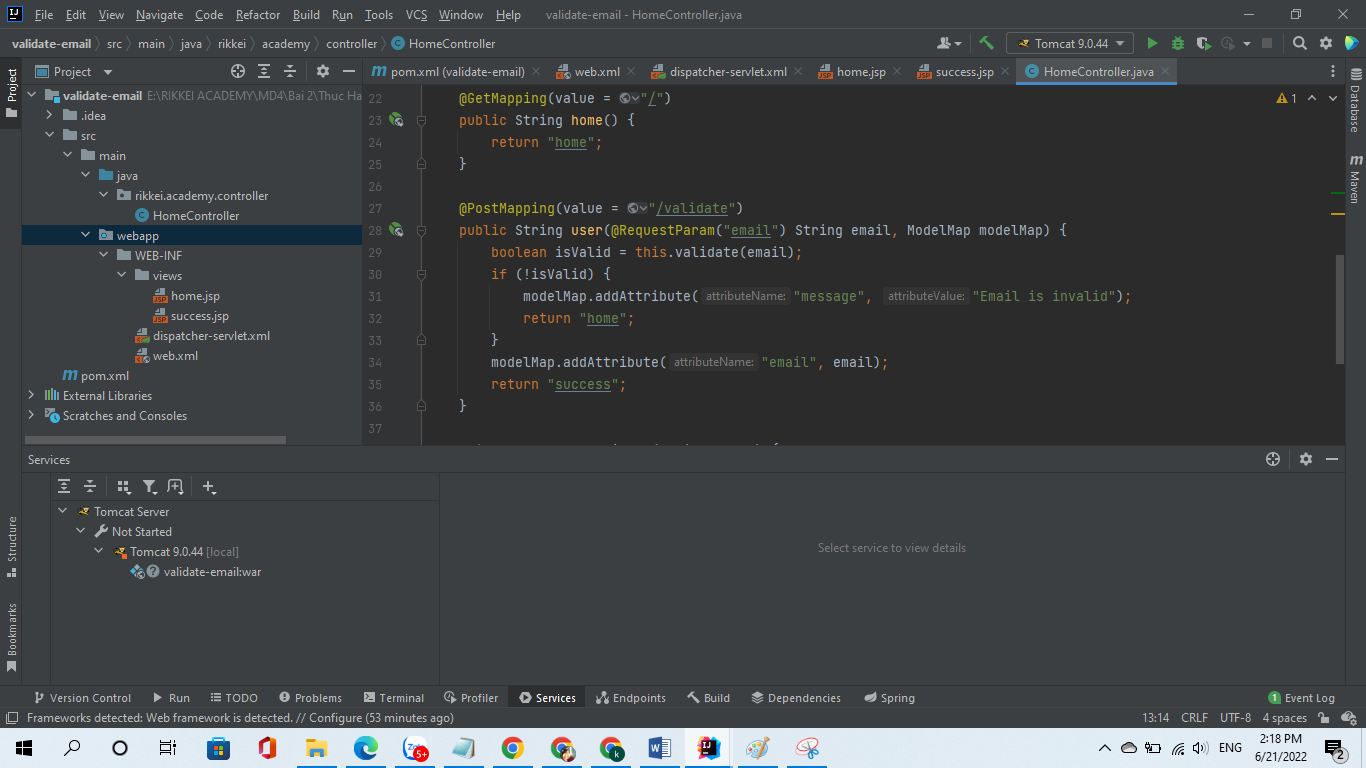
<bean id="viewResolver"  
 p:prefix="/WEB-INF/views/" p:suffix=".jsp"  
 class="org.springframework.web.servlet.view.InternalResourceViewResolver"/>

Đang tạo đường dẫn đến thư mục views chứa các file .jsp vì vậy ta sẽ tạo ra thư mục view và tạo các file jsp ở trong đó.



* + Tạo file: home.jsp và thêm code vào:
* <%@ **page** contentType="text/html;charset=UTF-8" language="java" %>  
  <html>  
  <head>  
   <title>Home</title>  
  </head>  
  <body>  
  <h1>Email Validate</h1>  
  <h3 style="color:red">${message}</h3>  
  <form action="validate" method="post">  
   <input type="text" name="email"><br>  
   <input type="submit" value="Validate">  
  </form>  
  </body>  
  </html>
  + Tạo file: **success.jsp** và thêm code vào:
* <%@ **page** contentType="text/html;charset=UTF-8" language="java" %>  
  <html>  
  <head>  
   <title>Success</title>  
  </head>  
  <body>  
  <h3>Hi ${email}</h3>  
  </body>  
  </html>

Xóa file: **index.jsp** có sẵn , add Tomcat và test kết quả với : email hợp lệ và không hợp lệ



Mã nguồn tham khảo:

<https://github.com/rikkei-academy/MD4-Bai2-TH1-Validate-Email-Spring-MVC-Maven.git>

### Hướng dẫn nộp bài:

Up code lên github.

Paste link github vào phần nộp bài.